Available online at www.sciencedirect.com

SCIENCE DIRECT®

Theoretical Computer Science 333 (2005) 355–371

Theoretical Computer Science

www.elsevier.com/locate/tcs

Time complexity of radio broadcasting: adaptiveness vs. obliviousness and randomization vs. determinism

Dariusz R. Kowalski[a, b, 1], Andrzej Pelc[c, *, 2]

[a] Instytut Informatyki, Uniwersytet Warszawski, Banacha 2, 02-097 Warszawa, Poland

[b] Max-Planck-Institut fuer Informatik, Stuhlsatzenhausweg 85, 66123 Saarbruecken, Germany

[c] Département d'informatique, Université du Québec en Outaouais, Hull, Qué. Canada J8X 3X7

Received 20 October 2003; received in revised form 1 March 2004; accepted 7 April 2004

Abstract

We consider the time of broadcasting in ad hoc radio networks modeled as undirected graphs. In such networks, every node knows only its own label and a linear bound on the number of nodes but is unaware of the topology of the network, or even of its own neighborhood. Our aim is to study to what extent the availability of two important characteristics of a broadcasting algorithm influences optimal broadcasting time. These characteristics are adaptiveness and randomization. Our contribution is establishing upper and lower bounds on optimal broadcasting time for three classes of algorithms: adaptive deterministic, oblivious randomized and oblivious deterministic. In two cases we present tight bounds, and in one case a small gap remains. We show that for deterministic adaptive algorithms time $\Omega(n)$ is required even for n-node networks of constant diameter. This lower bound is strongest possible, since linear time algorithms are known, and hence establishes optimal time $\Theta(n)$ for this class. For oblivious randomized algorithms we show an upper bound $\mathcal{O}(n \min\{D, \log n\})$ and a lower bound $\Omega(n)$ on optimal expected broadcasting time in n-node networks of diameter D. Finally, for

* Corresponding author.

E-mail addresses: darek@mimuw.edu.pl (D.R. Kowalski), pelc@uqo.ca (A. Pelc).

[1] A part of this work was done during this author's stay at the Research Chair in Distributed Computing of the Université du Québec en Outaouais, as a postdoctoral fellow. Partially supported by NSF-NATO grant 0209588 and by KBN grant 4T11C04425.

[2] Research supported in part by NSERC grant OGP 0008136 and by the Research Chair in Distributed Computing of the Université du Québec en Outaouais.

doi:10.1016/j.tcs.2004.04.017

oblivious deterministic algorithms we show matching upper and lower bounds $\Theta(n \min\{D, \sqrt{n}\})$ on optimal broadcasting time. Our results imply that enforcing obliviousness has at least as strong negative impact on broadcasting time as enforcing determinism, and that algorithms having both these features are strictly less efficient than those having only one of them.

Keywords: Radio broadcasting; Adaptiveness; Randomization

1. Introduction

A radio network is modeled as an *undirected* connected graph whose nodes are stations equipped with transmitter–receiver devices. Two nodes are adjacent if the transmitter of one of them can reach the other. Similarly as many papers in the domain of radio communication [3,7–12,14], we assume that nodes send messages in synchronous *steps* (time slots) controlled by a global clock. In every step every node acts either as a *transmitter* or as a *receiver*. A node acting as a transmitter sends a message which can potentially reach all of its neighbors. A node acting as a receiver in a given step gets a message, if and only if, exactly one of its neighbors transmits in this step. If at least two neighbors v and v' of u transmit simultaneously in a given step, none of the messages is received by u in this step. In this case we say that a *collision* occurred at u. It is assumed that the effect at node u of more than one of its neighbors transmitting is the same as that of no neighbor transmitting, i.e., a node cannot distinguish a collision from silence.

We consider ad hoc radio networks. In such networks, every node knows only its own label and a linear bound N on the number of nodes but is unaware of the topology of the network, or even of its own neighborhood. Nodes have distinct labels which are integers from the set $\{1, \ldots, N\}$.

One of the fundamental tasks in network communication is *broadcasting*. Its goal is to transmit a message from one node of the network, called the *source*, to all other nodes. Remote nodes get the source message via intermediate nodes, along paths in the network. We concentrate on one of the most important and widely studied performance parameters of a broadcasting algorithm, which is the *time* of broadcasting, defined as the number of steps used by the algorithm to inform all the nodes of the network. In the case of randomized algorithms, we are interested in expected broadcasting time. We assume that nodes can make *spontaneous* transmissions (even before obtaining the source message). This means that all nodes start the execution of the protocol simultaneously, and if the protocol calls for a transmission by a node that has not yet got the source message then this node transmits a control message. This capability can be used, e.g., to send some control messages, in order to perform preprocessing prior to broadcasting itself.

Among broadcasting algorithms using spontaneous transmissions there is a class of natural algorithms which we call *oblivious schemes*. For these algorithms, the decision whether a node transmits in a given step (or the probability of transmission in the case of randomized algorithms) depends only on the label of the node and on the step number (the scheme is oblivious of communication history). Oblivious schemes for gossiping (all-to-all communication) were considered, e.g., in [8] and fault-tolerant oblivious broadcasting was considered in [12]. Oblivious schemes are particularly easy to implement, as they require very little computation power to schedule transmissions, hence nodes of the network can be much simpler and cheaper devices.

The aim of this paper is to study the impact of two important characteristics of broadcasting algorithms on the time of optimal broadcasting. These characteristics are adaptiveness and randomization. In adaptive algorithms, as opposed to the above-defined oblivious schemes, the decision whether a node transmits in a given step depends on previously obtained messages, in addition to the label of the node and the step number. In such algorithms, a node may need to compute its decision of whether to transmit in a given step, using as input all previously obtained information. Adaptive algorithms are usually more efficient but, as mentioned above, not as easy to implement as oblivious schemes. Also, local computations may contribute to the hidden cost of such algorithms. The second characteristic studied in this paper is randomization. While randomized algorithms are usually faster than deterministic ones, the obvious drawback is that their time guarantees concern only the expected value and need not hold in all cases, as opposed to deterministic algorithms. Hence both adaptiveness and randomization can be viewed as two ways of speeding up the broadcasting process, each of these ways coming at some cost to the user. We want to analyze quantitatively the impact of each of these two features on broadcasting time.

1.1. Our results

The two characteristics that we intend to study, yield the following four types of broadcasting algorithms: adaptive randomized, adaptive deterministic, oblivious randomized and oblivious deterministic. In Section 1.2 we report what is known about broadcasting time for the first class, that of adaptive randomized algorithms. The contribution of the present paper is to establish upper and lower bounds on optimal broadcasting time for each of the remaining three classes. In two cases we present tight bounds, and in one case a small gap remains.

Our first result is a lower bound for deterministic adaptive algorithms. We show that time $\Omega(n)$ is required even for some n-node networks of constant diameter. This lower bound is strongest possible, since a linear time deterministic algorithm was presented in [7]. Hence we establish optimal time $\Theta(n)$ for this class. This lower bound was previously claimed in [3] (even for a stronger model in which every node knows its immediate neighborhood) but, as we proved in [18], the argument from [3] is incorrect, and in fact the result itself (claimed for the above stronger model) is false. Nevertheless, we show here that a different argument establishes the linear lower bound in our present model: for every deterministic broadcasting algorithm we construct a network of diameter at most 4 for which this algorithm requires time $\Omega(n)$ to broadcast.

For the case of oblivious randomized algorithms we establish an upper bound $\mathcal{O}(n \min\{D, \log n\})$ and a lower bound $\Omega(n)$ on expected broadcasting time in n-node networks of diameter D. Finally, for oblivious deterministic algorithms we show a lower bound $\Omega(n \min\{D, \sqrt{n}\})$ on broadcasting time, which matches the upper bound $\mathcal{O}(n \min\{D, \sqrt{n}\})$ following from [8]. Table 1 summarizes our results.

Table 1
Upper and lower bounds for execution time of different classes of algorithms (in the case of randomized algorithms this is expected time). We denote by $*$ the results from this paper

		Randomized	Deterministic
Adaptive	Upper b.	$\mathcal{O}(D \log \frac{n}{D} + \log^2 n)$ [20,13]	$\mathcal{O}(n)$ [7]
	Lower b.	$\Omega(D + \log^2 n)$ [1]	$\Omega(n)$ $*$
Oblivious	Upper b.	$\mathcal{O}(n \min\{D, \log n\})$ $*$	$\mathcal{O}(n \min\{D, \sqrt{n}\})$ [8]
	Lower b.	$\Omega(n)$ $*$	$\Omega(n \min\{D, \sqrt{n}\})$ $*$

Our results imply that enforcing obliviousness has at least as strong negative impact on broadcasting time as enforcing determinism, and that algorithms having both these features are strictly less efficient than those having only one of them. In our complexity analysis of algorithms we take strong advantage of the fact that networks are undirected (equivalently symmetric oriented). On the other hand, our lower bound proofs hold for arbitrary oriented networks as well.

1.2. Related work

Most of the results on broadcasting in radio networks can be divided into two parts: those which assume complete knowledge of the topology of the network at all nodes, or equivalently, dealing with centralized broadcasting for a given network, and those assuming only limited knowledge of the network at all nodes and dealing with distributed broadcasting in arbitrary networks.

Deterministic centralized broadcasting assuming complete knowledge of the network was considered in [6], where a $\mathcal{O}(D \log^2 n)$-time broadcasting algorithm was given for all n-node networks of diameter D. In [16], $\mathcal{O}(D + \log^5 n)$-time broadcasting was proposed. On the other hand, in [1] the authors proved the existence of a family of n-node networks of radius 2, for which any broadcast requires time $\Omega(\log^2 n)$.

One of the first papers to study deterministic distributed broadcasting in radio networks whose nodes have only limited knowledge of the topology, was [3]. The authors assumed that nodes know only their own label and labels of their neighbors. Under this scenario, a simple linear-time broadcasting algorithm based on DFS follows from [2]. In [3], the authors constructed a class of n-node graphs of radius 2, and claimed that every broadcasting algorithm requires time $\Omega(n)$ on one of these graphs. Unfortunately, due to a subtle error in the argument in [3] (cf. also [4]), this result is incorrect. Indeed, in [18] we constructed an algorithm that broadcasts in logarithmic time on all graphs from [3]. In this paper we show that a different argument establishes the linear lower bound in our present model, where every node knows only its own label but not labels of its neighbors. Many authors [5,7,9–11,14] studied deterministic distributed broadcasting in radio networks under this weaker assumption. In [7] the authors gave a broadcasting algorithm working in time $\mathcal{O}(n)$ for arbitrary n-node networks, assuming—as we do in the present paper—that nodes can transmit spontaneously, before getting the source message. On the other hand, in [5] a lower bound $\Omega(D \log n)$ on deterministic broadcasting time was proved for n-node networks of diameter D, if spontaneous transmissions are not allowed.

In [7,9,10,14,19,13] the model of directed graphs was used. Increasingly faster broadcasting algorithms working on arbitrary n-node (directed) radio networks were constructed, the currently fastest being the $\mathcal{O}(n \log^2 D)$-time algorithm from [13]. (Here D is the radius of the network, i.e, the longest distance from the source to any other node.) On the other hand, in [11] a lower bound $\Omega(n \log D)$ on broadcasting time was proved for directed n-node networks of radius D.

Randomized broadcasting algorithms in radio networks were studied, e.g., in [3,8,13,20,21]. For these algorithms, no topological knowledge of the network was assumed. In [3] the authors showed a randomized broadcasting algorithm running in expected time $\mathcal{O}(D \log n + \log^2 n)$. Unlike our randomized broadcasting algorithm, the algorithm from [3] is adaptive. In [20] we improved this upper bound by presenting a broadcasting algorithm with expected time $\mathcal{O}(D \log(n/D) + \log^2 n)$. (Shortly later, a similar result was obtained independently in [13].) The best lower bound known for adaptive randomized broadcasting time with the possibility of spontaneous transmissions is $\Omega(D + \log^2 n)$ from [1]. It should be noted that the lower bound $\Omega(D \log(n/D))$ from [21] assumes that spontaneous transmissions are precluded.

Oblivious algorithms (both deterministic and randomized) for the task of gossiping (all-to-all broadcasting) were considered in [8]. In particular, the authors showed an oblivious deterministic gossiping scheme working in time $\mathcal{O}(n^{3/2})$. This implies the same upper bound for the time of oblivious deterministic broadcasting. Deterministic oblivious fault-tolerant algorithms for broadcasting in radio networks were considered in [12].

2. Adaptive deterministic broadcasting

The main result of this section is a lower bound on the time of deterministic broadcasting (holding even when spontaneous transmissions are allowed). Given any deterministic broadcasting algorithm, we show that, for some networks of diameter 4, this algorithm requires linear time for broadcasting. This result has been previously claimed in [3] (even for a stronger model) but, as we showed in [18], the argument from [3] is incorrect, and in fact the result itself is false in this stronger model. We now show that (under our present model) this lower bound can be derived correctly.

The idea of the proof is the following. We construct the network step-by-step, using consecutive steps of the fixed broadcasting algorithm $\mathcal{A}$, and *assuming* that particular nodes got particular messages in given steps. In order to express this, we use the notion of *abstract history* of a node, formally defined below. Intuitively, an abstract history of a node v at a given step k consists of a sequence of messages received by this node until step k. Since the network is not yet constructed, it is not yet known which abstract history will become the real one—the one given by algorithm $\mathcal{A}$ running on the final network. We can ensure that, if a given node had some abstract history up to a certain step, then it would behave in a given way (this is captured by the notion of abstract action function, defined below). Based on that we do the next step of the construction of the network, and simultaneously define abstract histories of nodes in this step. These abstract histories are defined so as to prevent some nodes in layer L_2 of the network from getting any message for a long time. When the construction is finished, we prove that if the algorithm $\mathcal{A}$ runs on the resulting network then the real histories of all nodes are identical to the abstract (assumed) ones, and consequently, some nodes of layer L_2 will indeed fail to receive the source message for $\Omega(n)$ steps.

We adopt the definitions of histories and action function (and of their abstract counterparts) from [18], except the restriction precluding spontaneous transmissions and except knowledge of neighborhoods, unavailable in our present model.

Histories and message format: H_k denotes the history of computation of algorithm $\mathcal{A}$ until the end of step k. This is the set $\{H_k(v) : v \in V\}$, where $H_k(v)$ is the history of computation at node v, until the end of step k. Technically we assume that $H_0(v) = \emptyset$. For any v and k, $H_k(v)$ is a sequence of *messages* $(M_1(v), M_2(v), \ldots, M_k(v))$. Messages are defined inductively, as follows. $M_1(v)$ is either the pair $(\emptyset, \emptyset)$, called the *empty message* (when no message is received), or the pair $(0, \mathit{source\_message})$ (sent by node 0), or the pair $(w, \emptyset)$ (sent by node $w \neq 0$). $M_l(v)$ (for $l = 2, \ldots, k$) is the empty message if node v did not get any message in step l. Otherwise, it is a pair consisting of:

- the label of node w from which node v received a message in step l,
- history $H_{l-1}(w)$.

Notice that we restrict attention to messages conveying the entire history of the transmitter. If a particular protocol requires transmitting specific information, the receiver can deduce this information from the received history, since programs of all nodes are the same. History $H_k(v)$ containing only empty messages is called the *empty history*.

Action function and sets of transmitters: Given algorithm $\mathcal{A}$, we denote by $\pi(v, H_{k-1}(v))$ the action of node v in step k, if its history until the end of step $k-1$ is $H_{k-1}(v)$. The values of the function π can be 1 or 0: if the value is 1, node v is sending the message $(v, H_{k-1}(v))$ in step k, otherwise it is receiving in step k. Under a fixed history H_{k-1}, we define the set of neighbors of v transmitting in step k as follows: $T_k(v) = \{w \in N_v : \pi(w, H_{k-1}(w)) = 1\}$, where N_v denotes a set of all neighbors of node v.

Abstract objects: Let $v \in V$. An abstract history $\hat{H}_k(v)$ of node v, is defined as a sequence $(\hat{M}_1(v), \hat{M}_1(v), \ldots, \hat{M}_k(v))$ of *abstract messages*. $\hat{M}_1(v) = M_1(v)$, and $\hat{M}_l(v)$, for $l > 0$, is either the empty message or is of the format $(w, \hat{H}_{l-1}(w))$, for some $w \in V$. Technically $\hat{H}_0(v) = H_0(v) = \emptyset$. Notice that, in general, abstract histories and abstract messages are not necessarily linked to any particular protocol.

We also define the *abstract action function* $\hat{\pi}(v, \hat{H}_{k-1}(v))$ as an extension of the action function π described above: if $\pi(v, \hat{H}_{k-1}(v))$ is defined for some v then $\hat{\pi}(v, \hat{H}_{k-1}(v)) = \pi(v, \hat{H}_{k-1}(v))$. Otherwise, $\hat{\pi}(v, \hat{H}_{k-1}(v)) = 0$. We now define sets of *abstract transmitters* to node v in step k: $\hat{T}_k(v) = \{w \in N_v : \hat{\pi}(w, \hat{H}_{k-1}(w)) = 1\}$.

Consider the following class C_n^* of graphs, defined in [3]. For any nonempty subset S of $\{1, \ldots, n\}$ and any nonempty subset R of $\{n+1, \ldots, 2n\}$, $G_{S,R} \in C_n^*$ is the graph (V, E) such that $V = \{0, 1, \ldots, n\} \cup R$ and $E = \{(0, i) : i = 1, \ldots, n\} \cup \{(i, j) : i \in S, j \in R\}$. As usual, 0 is the source and L_0, L_1, L_2 denote the layers of this graph (L_i is the set of nodes at distance i from the source).

Let $\mathcal{T}$ denote any finite sequence of subsets of $\{0, 1, \ldots, 2n\}$. We will use the following procedure modifying some set $S \subseteq \{1, \ldots, 2n\}$:

Procedure MODIFY$(S, \mathcal{T})$
set `stop` $:= 0$
while `stop` $= 0$ **do**
- `stop` $:= 1$
- if there is a set $T_l \in \mathcal{T}$ such that $|T_l \cap S| = 1$, choose such a set with smallest index, say T_k, such that $T_k \cap S = \{i\}$; remove node i from S;
  set `stop` $:= 0$

2.1. Construction

Fix any broadcasting algorithm $\mathcal{A}$. We construct a graph $G_{S,R}$ from the class C_n^*, such that $\mathcal{A}$ requires time $\Omega(n)$ to broadcast on $G_{S,R}$. The construction is step-by-step, following consecutive steps of algorithm $\mathcal{A}$. We start the construction by initializing $S_0 = \{1, \ldots, n\}$, $R_0 = \{n + 1, \ldots, 2n\}$. Each node $v \in \{0\} \cup R_0$ has empty abstract history $\hat{H}_0(v)$, and for each node v in S_0, $\hat{M}_0(v) = (0, \textit{source message})$. In step k we will construct sets $S_k \subseteq \{1, \ldots, n\}$ and $R_k \subseteq \{n + 1, \ldots, 2n\}$, and construct abstract histories $\hat{H}_k(v)$. Finally, S will be S_{n-1}, and R will be R_{n-1}.

Our goal is to preserve the property, that after step k, none of the nodes in S_k has received any message from nodes in R_k and vice versa. We will preserve the following invariant after step k of the construction.
1. For every set $\hat{T}_l$, $l \leqslant k$, $|\hat{T}_l \cap S_k| \neq 1$ and $|\hat{T}_l \cap R_k| \neq 1$.
2. At least $n - |S_k|$ sets $\hat{T}_l$ are disjoint with S_k and at least $n - |R_k|$ sets $\hat{T}_l$ are disjoint with R_k, for $l \leqslant k$.
3. $|S_k| \geqslant n - k$ and $|R_k| \geqslant n - k$.
4. If $v \in R_k$ then $\hat{H}_k(v)$ is the empty history.

We describe step $k + 1 < n$ of the construction, assuming the validity of the invariant after step k.

Construction of sets S_{k+1} and R_{k+1}: Suppose that we constructed sets S_k and R_k, and each node v in network G_{S_k,R_k} has fixed abstract histories $\hat{H}_l(v)$, for all $l \leqslant k$ and $v \in V$. Let $\hat{T}_l = \bigcup_v \hat{T}_l(v)$ denote the set of all abstract transmitters in steps l, for $l \leqslant k + 1$, under fixed abstract histories $\hat{H}_l$. Let $\mathcal{T} = \{\hat{T}_l : l \leqslant k\}$.
1. Set $S := S_k$ and $R := R_k$.
2. Apply Procedure MODIFY$(S, \mathcal{T})$ to modify S; Apply Procedure MODIFY$(R, \mathcal{T})$ to modify R.
3. Set $S_{k+1} := S$ and $R_{k+1} := R$.

Construction of abstract history $\hat{H}_{k+1}(v)$: We define abstract history $\hat{H}_{k+1}(v)$ as abstract history $\hat{H}_k(v)$ concatenated with the empty abstract message, if $v \in \hat{T}_{k+1}$, and with the following abstract message $\hat{M}_{k+1}(v)$ otherwise:
- if $v \in R_{k+1}$ then $\hat{M}_{k+1}(v)$ is empty;
- if $v \in L_1$ and $\hat{T}_{k+1}(v) = \{0\}$ then $\hat{M}_{k+1}(v) = (0, \hat{H}_k(0))$;
- if $v \in L_1$ and $\hat{T}_{k+1}(v) \neq \{0\}$ then $\hat{M}_{k+1}(v)$ is the empty message;
- for $v = 0$, if $\hat{T}_{k+1}(0) = \{w\}$ then $\hat{M}_{k+1}(0) = (w, \hat{H}_k(w))$, otherwise $\hat{M}_{k+1}(0)$ is the empty message.

We continue the construction until step $k = n - 1$.

2.2. Analysis

The analysis is conducted as follows. First we prove, by induction on k, that the invariant holds after step k, and hence that the construction of $G_{\mathcal{A}}$ is correct. Then we show that the abstract history and the actual one on the constructed network $G_{\mathcal{A}}$ are identical. This implies that the algorithm takes linear time to broadcast on $G_{\mathcal{A}}$.

Lemma 1. *The invariant after step k of the construction holds, for* $k = 0, \ldots, n-1$.

Proof. Induction on step k of the construction. After step 0, the invariant is obvious. Suppose it holds after step $k \leqslant n-2$. Consider part 2 of the construction, corresponding to the set S.

Case 1: If $|S_k \cap \hat{T}_{k+1}| \neq 1$, then $S_{k+1} = S_k$ and $|S_k| \geqslant n-k > n-(k+1)$. This follows from the invariant after step k. There is no set $\hat{T}_l$ having a singleton intersection with S_{k+1}. The number of sets $\hat{T}_l$, for $l \leqslant k+1$, such that $\hat{T}_l \cap S_{k+1} = \emptyset$ is at least $n - |S_{k+1}|$, by part 2 of the invariant after step k.

The same properties hold for set R_{k+1}, if $|R_k \cap \hat{T}_{k+1}| \neq 1$.

Case 2: Suppose that $S_k \cap \hat{T}_{k+1} = \{v\}$. Then the set S becomes $S \setminus \{v\}$ in the beginning of Procedure MODIFY. During the remaining part of Procedure MODIFY, a modified set S can have a singleton intersection at most $|S_k|$ times, by part 2 of the invariant after step k. Each singleton intersection causes decreasing set S by one element, and the number of sets $\hat{T}_l$ having nonempty intersection with the modified set S decreases also by 1. Hence the invariant after step $k+1$ of the construction holds for set S_{k+1}, which is the last modified set S in this step. The same arguments apply to set R_{k+1}.

The property that for every $v \in R_k$, $\hat{H}_k(v)$ is the empty history, holds by the definition of abstract history in step $k+1$ of the construction. $\square$

Lemma 2. *If* $v \in G_{S_{n-1},R_{n-1}}$ *then* $\hat{H}_k(v) = H_k(v)$ *for all* $k = 0, \ldots, n-1$.

Proof. Induction on step k of the construction. For $k = 0$ the equality holds. Suppose that, for all $l \leqslant k$ and for all $v \in G_{S_{n-1},R_{n-1}}$, $\hat{H}_l(v) = H_l(v)$. We prove that $\hat{M}_{k+1}(v) = M_{k+1}(v)$ for all $v \in G_{S_{n-1},R_{n-1}}$. Notice that $\hat{T}_l(v) = T_l(v)$ holds for every $l \leqslant k+1$ and $v \in G_{S_{n-1},R_{n-1}}$, since these sets depend only on histories $\hat{H}_l$ and H_l, for $l \leqslant k$, which are equal by the inductive assumption. It follows that $\hat{T}_l = T_l$. Consequently, if $v \in \hat{T}_{k+1}$ then $v \in T_{k+1}$, and $\hat{M}_{k+1}(v) = M_{k+1}(v)$ is the empty message. Suppose $v \notin \hat{T}_{k+1}$.

- If $v \in R_{n-1} \subseteq R_{k+1}$ then $\hat{M}_{k+1}(v)$ is empty. Suppose that there is a node $w \in S_{n-1} \subseteq S_{k+1}$ such that $T_{k+1}(v) = \{w\}$. This means that $\hat{T}_{k+1} \cap S_{n-1} = T_{k+1} \cap S_{n-1} = T_{k+1}(v) = \{w\}$, which contradicts the invariant after step $n-1$. Hence $M_{k+1}(v)$ is the empty message.
- If $v \in L_1$ and $\hat{T}_{k+1}(v) = \{0\}$ then $\hat{M}_{k+1}(v) = (0, \hat{H}_k(0))$. It follows that $T_{k+1}(v) = \hat{T}_{k+1}(v) = \{0\}$ and $M_{k+1}(v) = (0, H_k(0)) = (0, \hat{H}_k(0))$.
- If $v \in L_1$ and $\hat{T}_{k+1}(v) \neq \{0\}$ then $\hat{M}_{k+1}(v)$ is the empty message. There are two cases.
  1. If $|\hat{T}_{k+1}(v)| \neq 1$, then $|T_{k+1}(v)| \neq 1$ and $M_{k+1}(v)$ is the empty message.
  2. If $\hat{T}_{k+1}(v) = \{w\}$, where $w \in R_{n-1}$, then $T_{k+1} \cap R_{n-1} = \{w\}$, which contradicts the invariant after step $n-1$ of construction. Hence the second case is not feasible.

- Suppose that $v = 0$. If $\hat{T}_{k+1}(0) = \{w\}$ then $\hat{M}_{k+1}(0) = (w, \hat{H}_k(w))$. In this case $T_{k+1}(0) = \hat{T}_{k+1}(0) = \{w\}$ and $M_{k+1}(0) = (w, H_k(w)) = (w, \hat{H}_k(w))$. If $|\hat{T}_{k+1}(0)| \neq 1$ then $|T_{k+1}(0)| \neq 1$ and consequently both $\hat{M}_{k+1}(0)$ and $M_{k+1}(0)$ are empty messages. $\square$

Theorem 1. *For every integer n and every broadcasting algorithm $\mathcal{A}$ there exists a network $G_{\mathcal{A}}$ of size at most $2n+1$ and diameter 4, such that algorithm $\mathcal{A}$ requires time $\Omega(n)$ to broadcast on $G_{\mathcal{A}}$.*

Proof. Let $G_{\mathcal{A}}$ be the graph $G_{S_{n-1},R_{n-1}}$. It has at most $2n+1$ nodes from range $\{0, 1, \ldots, 2n\}$, and it has diameter 4, since sets S_{n-1} and R_{n-1} are nonempty, by Lemma 1. By Lemmas 1 and 2, the history of each node in R_{n-1} is empty until step $n-1$ of algorithm $\mathcal{A}$, which completes the proof. $\square$

3. Oblivious randomized broadcasting

3.1. The lower bound

In this section we prove the lower bound $\Omega(n)$ on the expected broadcasting time of oblivious randomized algorithms working in n-node networks. Denote by $p_i(v)$ the probability of transmission by node v in step i. Let $p_i = \sum_{v=1}^{n} p_i(v)$.

Theorem 2. *For every oblivious randomized broadcasting algorithm $\mathcal{A}$ and every sufficiently large n, there exists an n-node network $G_{\mathcal{A}}$ of diameter 3, such that the algorithm $\mathcal{A}$ requires time $\Omega(n)$, with probability at least 1/2, to complete broadcasting on $G_{\mathcal{A}}$.*

Proof. We use $[m]$ to denote the set $\{1, \ldots, m\}$, for any positive integer m. Assume that n is sufficiently large. For simplicity assume also that n is even. Let $\mathcal{A}$ be any oblivious randomized algorithm. Define the graph $G_{\mathcal{A},v}$, for any $v \in [n-2]$, as a graph on the set $\{0\} \cup [n-1]$ of nodes with the following set E of edges. $E = \{\{w, n-1\} : w \in [n-2]\} \cup \{\{0, v\}\}$ (see Fig. 1). By definition, the graph $G_{\mathcal{A},v}$ has diameter 3. We will show that there exists a node v such that during the execution of algorithm $\mathcal{A}$ on graph $G_{\mathcal{A},v}$, node $n-1$ does not receive the source message before time $(n-2)/2$, with probability at least 1/2.

In view of the obliviousness of algorithm $\mathcal{A}$, for every node w and for every step i, there are fixed probabilities $p_i(w)$ that node w transmits in step i. Observe that

$$\sum_{w\in[n-2]} p_i(w) \prod_{z\neq w, z\in[n-2]} (1 - p_i(z)) \leqslant 1$$

is the probability that some single node in $[n-2]$ transmits in step i. It follows that

$$\sum_{i=1}^{(n-2)/2} \sum_{w=1}^{n-2} p_i(w) \prod_{z\neq w, z\in[n-2]} (1 - p_i(z)) \leqslant \frac{n-2}{2}.$$

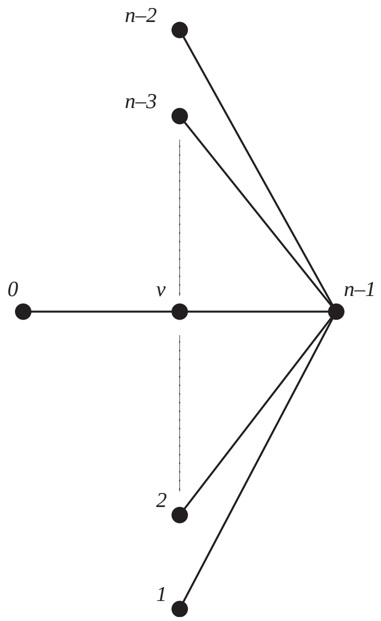

Fig. 1. Network $G_{\mathcal{A},v}$.

Hence there exists $v \in [n-2]$ such that

$$\sum_{i=1}^{(n-2)/2} p_i(v) \prod_{z \neq v, z \in [n-2]} (1 - p_i(z)) \leqslant \frac{1}{2}. \tag{1}$$

Define $G_{\mathcal{A}} = G_{\mathcal{A},v}$. Let X_i be the random variable equal 1 if v transmits in step $i \leqslant (n-2)/2$ and every $w \in [n-2] \setminus \{v\}$ does not transmit in step i, and let X_i be equal 0 otherwise. Notice that, by inequality 1, we get $\mathsf{E}\left[\sum_{i=1}^{(n-2)/2} X_i\right] \leqslant \frac{1}{2}$. Using Markov inequality we obtain $\Pr\left[\sum_{i=1}^{(n-2)/2} X_i \geqslant 1\right] \leqslant \frac{1}{2}$. The event $\sum_{i=1}^{(n-2)/2} X_i < 1$, occurring with probability at least 1/2, means that node $n-1$ does not receive the source message from v by step $(n-2)/2$. $\square$

Corollary 1. *For all parameters n and $3 \leqslant D < n$, and for any oblivious randomized algorithm $\mathcal{A}$, there exists an n-node network $G_{\mathcal{A}}$ of diameter D, such that the expected broadcasting time of algorithm $\mathcal{A}$ on network $G_{\mathcal{A}}$ is $\Omega(n)$.*

3.2. Oblivious randomized algorithm

Before presenting our oblivious randomized broadcasting algorithm, we describe its general idea and compare it to the adaptive randomized algorithm from [3]. Both algorithms use the same framework: a "competition" procedure is repeated several times, the aim of which is to guarantee that exactly one of competing nodes transmits in some round, with positive probability. There is, however, an important difference between the adaptive scenario from [3] and our case of oblivious algorithms. In the adaptive case the "competition" used the following k-round procedure Decay. In each round, competing nodes transmit. At the end of each round, each competing node flips a symmetric coin. Those nodes that get tails do not compete in the following rounds.

In our oblivious scenario, we cannot use procedure Decay, since actions of nodes cannot be modified on-line. Hence we arrange competitions differently: in the lth round all nodes transmit with the same probability but this probability decreases by a factor of 1/2 in the next round. Another difference is that we need to repeat the competition more times than in [3]. A similar idea has been also used in [8] in the context of oblivious randomized gossiping.

Here is the description of our broadcasting algorithm. Recall that N is an upper bound on all labels, $N = \mathcal{O}(n)$.

Algorithm Randomized-Oblivious

$count := 1$
repeat $\lceil N^2/\log N \rceil$ times
 for $l := 1$ to $\lceil \log N \rceil$ do
  (a) each node transmits independently with probability $1/2^l$
  (b) node with label *count* transmits, $count := count + 1 \bmod N$

Theorem 3. *Algorithm Randomized-Oblivious completes broadcasting on any n-node network of diameter D, with probability at least* $1 - 1/n^2$, *in time* $\mathcal{O}(n \, \min\{D, \log n\})$.

Proof. First observe that if $D < \log n$ then performing only steps (b) in the for-loop completes broadcasting in time $\mathcal{O}(N \cdot D) = \mathcal{O}(n \cdot D)$.

Suppose $D \geqslant \log n$ and consider both steps (a) and (b) of the algorithm. We define a *stage* as one execution of the entire loop "for $l := 1$ to $\lceil \log N \rceil$". Consider a shortest path from node 0 to node v consisting of consecutive nodes $v_0 = 0, \ldots, v_k = v$, where $k \leqslant D$.

Let d_i denote the degree of node v_i. Notice that $\sum_{i=1}^{k} d_i \leqslant 2n$.

Claim 1. *Node v_i receives a message from v_{i-1}, during any stage j with probability at least* $1/8d_i$.

Proof. For $d_i = 1$ this is obvious—in the execution of the loop for $l = 1$ the transmission is successful with probability 1/4. Assume $d_i > 1$. Consider the execution of the loop "for" in stage j for variable $l = \lceil \log d_i \rceil$. The probability that v_i receives a message from v_{i-1} in lth execution of loop "for" during stage j is at least

$$\frac{1}{2^l} \cdot \left(1 - \frac{1}{2^l}\right)^{d_i - 1} \cdot \left(1 - \frac{1}{2^l}\right) \geqslant \frac{1}{2d_i} \cdot \left(1 - \frac{1}{d_i}\right)^{d_i} \geqslant \frac{1}{8d_i}.$$

This completes the proof of the claim. □

Let t_i denote the first stage in the execution of the algorithm when node v_i receives the source message from node v_{i-1}. It follows from Claim 1 that $t_i - t_{i-1} = \mathcal{O}(d_i)$ with constant probability. We have $t_k = \sum_{i=1}^{k}(t_i - t_{i-1})$, where $t_0 = 0$. Notice that t_k is an upper bound on the time when node v_k gets the source message. For every stage $j = 1, \ldots, t_k$ let $\alpha(j)$ be the smallest number $i = 1, \ldots, k$ such that node v_i has not received the source message from v_{i-1} by the beginning of stage j. Partition all stages $j = 1, \ldots, t_k$ into sets A and C as follows: if $d_{\alpha(j)} \leqslant n/\log n$ then $j \in A$, and if $d_{\alpha(j)} > n/\log n$ then $j \in C$.

Counting $|A|$. Let W contain all nodes v_i in the path such that $d_i \leqslant n/\log n$. First note that if $W \leqslant \log n$ then $|A| = \mathcal{O}(n)$. Indeed, considering only the round-robin process in steps (b), we get that the value of $\alpha(j)$ changes after at most N steps, that is after at most $N/\log N = \mathcal{O}(n/\log n)$ stages; multiplying it by $|W|$ we get the upper bound $\mathcal{O}(n)$ on the size of A.

Suppose that $|W| > \log n$. For every $a = 1, \ldots, \log(n/\log n)$, let p_a denote the total number of nodes $v_i \in W$ such that $2^{a-1} < d_i \leqslant 2^a$, and let P_a denote the set of such nodes v_i.

Note that if $d_i = 1$ then $i = k$, and consequently if the neighbor v_{k-1} of node v_k has the source message then v_k gets the source message in $\mathcal{O}(\log n)$ next stages, with probability at least $1/n^4$. Consider the case when $d_i > 1$.

Claim 2. *If* $v_{\alpha(j)} \in P_a$, *for some* $a = 1, \ldots, \log(n/\log n)$, *then the probability that* $v_{\alpha(j+2^{a+4}-1)} = v_{\alpha(j)}$ *is at most* $1/\mathrm{e}$.

Proof. By Claim 1, node $v_{\alpha(j)}$ gets the source message in one stage with probability at least $1/(8d_{\alpha(j)})$. These events are independent for different stages. Consequently, the probability that during 2^{a+4} stages node $v_{\alpha(j)}$ does not get the source message is at most

$$\left(1 - \frac{1}{8d_{\alpha(j)}}\right)^{2^{a+4}} \leqslant (1 - 2^{-a-4})^{2^{a+4}} \leqslant 1/e. \qquad \square$$

Let A_a, for $a = 1, \ldots, \log(n/\log n)$, denote the set of those stages j for which $v_{\alpha(j)} \in P_a$.

Claim 3. $|A_a| \leqslant 2e2^{a+4}(p_a + 16\log n)$, *with probability at least* $1 - 1/n^4$.

Proof. For each node $v_i \in P_a$, index i may be the value of function α for at most 2^{a+4} stages, with probability at least $1 - 1/\mathrm{e}$, in view of Claim 2. Consider random variables X_j (corresponding to stages $(j-1) \cdot 2^{a+4}, \ldots, j2^{a+4} - 1$): $X_j = 1$ if $\alpha(j2^{a+4} - 1) > \alpha((j-1)2^{a+4})$, and $X_j = 0$ otherwise. Note that random variables X_j are not independent, but since the conditional probability $\Pr[X_j = 1|\Psi]$, for any computational history Ψ before stage $(j-1)2^{a+4}$, is at least $1 - 1/\mathrm{e}$, we can stochastically lower-bound each variable X_j by a random Bernoulli trial Y_j with probability of success $1 - 1/\mathrm{e}$. Hence Chernoff bound applied to variables Y_j implies that the probability of the event that the number of successful transmissions during $2\mathrm{e}2^{a+4}(p_a + 16\log n)$ stages is less than p_a, is at most $\mathrm{e}^{-(p_a+16\log n)/4} \leqslant 1/n^4$ (note that the expected number of successful transmissions in $2\mathrm{e}2^{a+4}(p_a + 16\log n)$ stages is at least $2(p_a + 16\log n)$). $\square$

By definition, $|A| = |A_1| + \cdots + |A_{\log(n/\log n)}|$. Hence, with probability at least $1 - \log n \cdot 1/n^4$,

$$\begin{aligned}
|A| &\leqslant \sum_{a=1}^{\log(n/\log n)} 2\mathrm{e}2^{a+4}(p_a + 12\log n) \\
&\leqslant 2\mathrm{e} \sum_{a=1}^{\log(n/\log n)} 2^{a+4} p_a + 24\mathrm{e} \sum_{a=1}^{\log(n/\log n)} 2^{a+4} \log n \\
&\leqslant 32\mathrm{e} \cdot 2 \sum_{i=1}^{k} d_i + 192\mathrm{e} \cdot 2^{\log(n/\log n)} \log n \leqslant 128\mathrm{e}n + 192\mathrm{e}n = \mathcal{O}(n),
\end{aligned}$$

where we used the facts that $2^a p_a$ is at most twice the sum of degrees of nodes in P_a, and that $\sum_{i=1}^{k} d_i \leqslant 2n$. Adding the special case when $d_i = 1$ we get again that $|A| = \mathcal{O}(n + \log n) = \mathcal{O}(n)$ with probability at least $1 - \log n/n^4 - 1/n^4$.

Counting $|C|$. Notice that analyzing only steps (b) we get that the value of $\alpha(j)$ changes after at most N steps (by the round-robin property), which is at most $N/\log N = \mathcal{O}(n/\log n)$ stages. By the counting argument we have that the number of nodes v_i on the path, such that $d_i > n/\log n$ is at most $2\log n$ (by the definition of C and inequality $\sum_{i=1}^{k} d_i \leqslant 2n$). Hence $|C| \leqslant \mathcal{O}(n/\log n) \cdot 2\log n = \mathcal{O}(n)$. All this happens with probability 1 (since we analyze the behavior of deterministic round-robin in steps (b)).

Counting $|A \cup C|$. Summarizing, the total number of stages in set $A \cup C$ is bounded by $\mathcal{O}(n)$, with probability at least $1 - (\log n + 1)/n^4 \geqslant 1 - 1/n^3$, and each stage takes $\mathcal{O}(\log n)$ steps. The total time is thus $\mathcal{O}(n \log n)$, with probability at least $1 - 1/n^3$. Since we need to consider all n possibilities for node v, we have broadcasting time $\mathcal{O}(n \log n)$, with probability at least $1 - n/n^3 = 1 - 1/n^2$. □

Using steps (b) of the algorithm in the event not covered by the above theorem (a total of $\Theta(N^2)$ such steps), guarantees completion of broadcasting with probability 1. The contribution of this event to the expected completion time is $O(N^2 \cdot 1/n^2) = O(1)$. Hence we get the following.

Corollary 2. *The expected broadcasting time of Algorithm Randomized-Oblivious is* $\mathcal{O}(n \min\{D, \log n\})$.

Moreover, our algorithm performs gossiping with probability at least $1 - 1/n$. Hence, similarly as above (now the contribution of the unlikely event to the expected completion time is $\mathrm{O}(n)$ instead of $\mathrm{O}(1)$), we get that the expected time of gossiping is $\mathcal{O}(n \min\{D, \log n\})$, thus improving the result from [8].

4. Oblivious deterministic broadcasting

We finally consider deterministic oblivious broadcasting schemes. For these algorithms, every node decides whether to transmit in a given step, depending only on its label and on the step number (the scheme is oblivious of communication history). This means that, for all $i = 1, 2, \ldots$, sets T_i of nodes transmitting in step i are fixed in advance.

We first prove a lower bound on the time of deterministic oblivious broadcasting. In [11] the authors proved a lower bound on the size of a combinatorial structure called a strongly selective family. This structure is also known in the literature as a superimposed code [17], or a cover free family [15]. We will use the following definition. A family $\mathcal{F}$ of subsets of R is called $(|R|, k)$-*strongly selective*, for $k \leqslant |R|$, if for every subset Z of R such that $|Z| \leqslant k$, and for every element $z \in Z$, there is a set $F \in \mathcal{F}$ such that $Z \cap F = \{z\}$.

Lemma 3 (*Clementi et al. [11]*). *Let $\mathcal{F}$ be an $(|R|, k)$-strongly selective family. Then*
(a) *if $3 \leqslant k < \sqrt{2|R|}$ then $|\mathcal{F}| \geqslant (k^2/48 \log k) \log |R|$,*
(b) *if $k \geqslant \sqrt{2|R|}$ then $|\mathcal{F}| \geqslant |R|$.*

Our lower bound is proved by constructing, for any deterministic oblivious broadcasting scheme $\mathcal{A}$, a network on which this scheme broadcasts slowly. Similarly as in [11], the idea is to choose a layered network. In both cases lower bounds on the size of combinatorial structures are used. There is, however, an important difference between our scenario of oblivious protocols and the scenario from [11], where protocols could be adaptive. Since adaptive protocols have more flexibility, the authors of [11] could only use a lower bound on the size of *selective families* (selectivity is a weaker requirement than strong selectivity, hence the size of selective families can be smaller than strongly selective ones). Moreover, they had to construct a directed graph forcing slow broadcasting, in order to impose one-way information flow. In our case, the assumption of obliviousness enables us to use the larger lower bound on the size of strongly selective families, and we can even construct an *undirected* layered graph on which the given protocol (oblivious broadcasting scheme) works slowly. This explains why the lower bound is larger in our case.

Construction. Let $\mathcal{A}$ be a fixed oblivious broadcasting scheme. We describe the construction of an n-node network of radius D, for $7 \leqslant D \leqslant \sqrt{n/8}$. The remaining ranges of D are simple to handle. (For $D < 7$ we use the construction from Theorem 1 and attach a simple path to get radius D, and for $D > \sqrt{n/8}$ the construction is as for $D = \sqrt{n/8}$ with a simple path attached to get radius D.)

In order to construct network $G_{\mathcal{A}}$, we first describe sets of nodes X_k and selected nodes $v_k \in X_k$, for $k \leqslant \lfloor D/2 \rfloor$. Let $X_0 = \{0\}$, $v_0 = 0$. Let $R_k = \{1, \ldots, n-1\} \setminus \bigcup_{i<k} X_i$. We will preserve the following invariant after step $k \leqslant \lfloor D/2 \rfloor$ of the construction:

- All sets $X_0, \ldots, X_k$ and nodes $v_0, \ldots, v_k$ are constructed.
- $|R_{k+1}| > n/2$ and none of nodes in R_{k+1} has received the source message by step $k \cdot \lfloor n/2 \rfloor$.

We now describe the construction of X_{k+1} and v_{k+1}. Sets T_i of nodes transmitting in step i, for all $i = 1, 2, \ldots$, are fixed before the construction.

Consider the family of sets $\{T_i \cap R_{k+1} : k \cdot \lfloor n/2 \rfloor < i \leqslant (k+1) \cdot \lfloor n/2 \rfloor\}$. Since this family has size $\lfloor n/2 \rfloor$ and $\lfloor n/(2D) \rfloor \geqslant \sqrt{2n} \geqslant \sqrt{2|R_{k+1}|}$, it cannot be a $(|R_{k+1}|, \lfloor n/(2D) \rfloor)$-strongly selective family, in view of the lower bound in Lemma 3 point (b). Consequently there is a nonempty set $X_{k+1} \subseteq R_{k+1}$ of size at most $\lfloor n/(2D) \rfloor$, and a node $v_{k+1} \in X_{k+1}$ such that $T_i \cap X_{k+1} \neq \{v_{k+1}\}$, for any i satisfying $k \cdot \lfloor n/2 \rfloor < i \leqslant (k+1) \cdot \lfloor n/2 \rfloor$.

Let $X = \bigcup_{k+1}^{\lfloor D/2 \rfloor} X_k$. We now describe the set E of edges between nodes in X. Let $L_0 = \{0\}$, $L_1 = \{v_1\}$, $L_k = \{v_k\} \cup X_{k-2} \setminus \{v_{k-2}\}$, for $k = 2, \ldots, \lfloor D/2 \rfloor$, $L_{\lfloor D/2 \rfloor + 1} =$

$X_{\lfloor D/2\rfloor -1} \setminus \{v_{\lfloor D/2\rfloor -1}\}$, $E = \bigcup_{k=1}^{\lfloor D/2\rfloor +1} \Big\{\{v_{k-1}, x\} : x \in L_k\Big\}$. Let $G^*_{\mathcal{A}}$ be the graph (X, E). Clearly, the above defined sets L_i are layers of $G^*_{\mathcal{A}}$. Consider a path P of length $D - \lfloor D/2\rfloor - 3$, consisting of elements of $R_{\lfloor D/2\rfloor}$, with all other elements of $R_{\lfloor D/2\rfloor}$ attached to one end of this path. Network $G_{\mathcal{A}}$ is defined by attaching the other end of P to some node in layer $L_{\lfloor D/2\rfloor +1}$ of $G^*_{\mathcal{A}}$. Notice that $|R_{\lfloor D/2\rfloor}| > (D - \lfloor D/2\rfloor - 3) + 1$, hence the path P with attached nodes is well defined. Consequently, the network $G_{\mathcal{A}}$ has n nodes and radius D.

This completes the construction of network $G_{\mathcal{A}}$. It remains to prove that algorithm $\mathcal{A}$ requires time $\Omega(n \min\{D, \sqrt{n}\})$ to broadcast on $G_{\mathcal{A}}$.

Theorem 4. *For all parameters n, D, and for any deterministic oblivious broadcasting scheme $\mathcal{A}$, there exists an n-node network $G_{\mathcal{A}}$* of radius $\Theta(D)$, *such that scheme $\mathcal{A}$ requires time $\Omega(n \min\{D, \sqrt{n}\})$ to broadcast on $G_{\mathcal{A}}$.*

Proof. Let $\mathcal{A}$ be a fixed oblivious broadcasting scheme. First observe that for $1 < D < 7$ the result follows from Theorem 1. If $D > \sqrt{n/8}$, the result is straightforward. Hence we may assume that $7 \leqslant D \leqslant \sqrt{n/8}$. For this range the previous construction was carried out.

We prove that the invariant holds for every $k \leqslant \lfloor D/2\rfloor$. The proof is by induction on k. After step $k = 0$, it is clear. Suppose that after step $k < \lfloor D/2\rfloor$ the invariant holds. We prove it for $k + 1$.

Set X_{k+1} and node v_{k+1} are well defined since sets $T_i \cap R_{k+1}$ are nonempty and Lemma 3 point (b) applies. By definition, set R_{k+2} is of size at least $n - \sum_{l \leqslant k+1} |X_l| > n - \lfloor D/2\rfloor \cdot \lfloor n/(2D)\rfloor > n/2$. Every node v in R_{k+2} is at distance at least $k + 2$ from the source. The only neighbor in L_{k+1} of nodes from L_{k+2} is v_{k+1}. We show that node v_{k+1} does not transmit successfully to layer L_{k+2} in steps i such that $k \cdot \lfloor n/2\rfloor < i \leqslant (k+1) \cdot \lfloor n/2\rfloor$. If it transmitted in some such step i, then we would have $T_i \cap X_{k+1} = \{v_{k+1}\}$, which contradicts Case 1 of the construction.

We finally prove a lower bound on execution time of algorithm $\mathcal{A}$ broadcasting on network $G_{\mathcal{A}}$, in Case 1. It is sufficient to observe that no node in layer $L_{\lfloor D/2\rfloor +1}$ receives the source message by step $(\lfloor D/2\rfloor + 1) \cdot \lfloor n/2\rfloor$. Consequently, broadcasting is not terminated by step $\Omega(nD)$, for $D \leqslant \sqrt{n/8}$.

This shows that algorithm $\mathcal{A}$ requires time $\Omega(n \min\{D, \sqrt{n}\})$ to broadcast on some n-node network of radius D. □

Notice that there exists a deterministic oblivious scheme performing broadcasting in any n-node network of diameter D in time $\mathcal{O}(n \min\{D, \sqrt{n}\})$. In [8], a deterministic oblivious broadcasting scheme, working in time $\mathcal{O}(n^{3/2})$ for arbitrary n-node networks, was proposed. By interleaving this scheme with the simple round robin (oblivious) algorithm, working in time $O(nD)$, we obtain the upper bound $\mathcal{O}(n \min\{D, \sqrt{n}\})$ on deterministic oblivious broadcasting time, matching the lower bound from Theorem 4.

5. Conclusion

We considered optimal broadcasting time for three classes of algorithms: adaptive deterministic, oblivious randomized and oblivious deterministic, assuming that spontaneous transmissions are allowed. For deterministic adaptive algorithms we established the lower bound $\Omega(n)$, thus proving that optimal time is $\Theta(n)$ for this class. For oblivious randomized algorithms we showed an upper bound $\mathcal{O}(n \ \min\{D, \log n\})$ and a lower bound $\Omega(n)$ on optimal expected broadcasting time in n-node networks of diameter D. Finally, for oblivious deterministic algorithms we showed matching upper and lower bounds $\Theta(n \min\{D, \sqrt{n}\})$ on optimal broadcasting time.

The main open problem left by the results of this paper is closing the gap between the upper and lower bounds on expected broadcasting time for oblivious randomized algorithms (this gap is a factor of $\Theta(\min\{D, \log n\})$). This adds to the previously open problem of closing an analogous gap for adaptive randomized algorithms with spontaneous transmissions ($\Theta(\log(n/D))$ in this case).

References

[1] N. Alon, A. Bar-Noy, N. Linial, D. Peleg, A lower bound for radio broadcast, J. Comput. System Sci. 43 (1991) 290–298.

[2] B. Awerbuch, A new distributed depth-first-search algorithm, Inform. Process. Lett. 20 (1985) 147–150.

[3] R. Bar-Yehuda, O. Goldreich, A. Itai, On the time complexity of broadcast in radio networks: an exponential gap between determinism and randomization, J. Comput. System Sci. 45 (1992) 104–126.

[4] R. Bar-Yehuda, O. Goldreich, A. Itai, Errata regarding On the time complexity of broadcast in radio networks: an exponential gap between determinism and randomization, available from http://www.wisdom.weizmann.ac.il/∼oded/p_ bgi.html.

[5] D. Bruschi, M. Del Pinto, Lower bounds for the broadcast problem in mobile radio networks, Distributed Comput. 10 (1997) 129–135.

[6] I. Chlamtac, O. Weinstein, The wave expansion approach to broadcasting in multihop radio networks, IEEE Trans. Comm. 39 (1991) 426–433.

[7] B.S. Chlebus, L. Gasieniec, A. Gibbons, A. Pelc, W. Rytter, Deterministic broadcasting in unknown radio networks, Distributed Comput. 15 (2002) 27–38.

[8] B.S. Chlebus, L. Gasieniec, A. Lingas, A. Pagourtzis, Oblivious gossiping in ad-hoc radio networks, in: Proc. of Fifth Internat. Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (DIALM'2001), 2001, pp. 44–51.

[9] B.S. Chlebus, L. Gąsieniec, A. Östlin, J.M. Robson, Deterministic radio broadcasting, in: Proc. of 27th Int. Coll. on Automata, Languages and Programming (ICALP'2000), Lecture Notes in Computer Science, Vol. 1853, 2000, pp. 717–728.

[10] M. Chrobak, L. Gąsieniec, W. Rytter, Fast broadcasting and gossiping in radio networks, in: Proc. of 41st Symp. on Foundations of Computer Science (FOCS'2000), 2000, pp. 575–581.

[11] A.E.F. Clementi, A. Monti, R. Silvestri, Selective families, superimposed codes, and broadcasting on unknown radio networks, in: Proc. of 12th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'2001), 2001, pp. 709–718.

[12] A.E.F. Clementi, A. Monti, R. Silvestri, Round robin is optimal for fault-tolerant broadcasting on wireless networks, in: Proc. of Ninth Annual European Symp. on Algorithms (ESA'2001), 2001, pp. 452–463.

[13] A. Czumaj, W. Rytter, Broadcasting algorithms in radio networks with unknown topology, in: Proc. of 44th Ann. Symp. on Foundations of Computer Science (FOCS'2003), 2003, pp. 492–501.

[14] G. De Marco, A. Pelc, Faster broadcasting in unknown radio networks, Inform. Process. Lett. 79 (2001) 53–56.

[15] P. Erdös, P. Frankl, Z. Füredi, Families of finite sets in which no set is covered by the union of r others, Israel J. Math. 51 (1985) 79–89.

[16] I. Gaber, Y. Mansour, Broadcast in radio networks, in: Proc. of Sixth Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'1995) 1995, pp. 577–585.

[17] W.H. Kautz, R.R.C. Singleton, Nonrandom binary superimposed codes, IEEE Trans. Information Theory 10 (1964) 363–377.

[18] D. Kowalski, A. Pelc, Deterministic broadcasting time in radio networks of unknown topology, in: Proc. of 43rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS'2002), 2002, pp. 63–72.
[19] D. Kowalski, A. Pelc, Faster deterministic broadcasting in ad hoc radio networks, in: Proc. of 20th Ann. Symp. on Theoretical Aspects of Computer Science (STACS'2003), Lecture Notes in Computer Science, Vol. 2607, 2003, pp. 109–120.
[20] D. Kowalski, A. Pelc, Broadcasting in undirected ad hoc radio networks, in: Proc. of 22nd ACM Symp. on Principles of Distributed Computing (PODC'2003) pp. 73–82.
[21] E. Kushilevitz, Y. Mansour, An $\Omega(D \log(N/D))$ lower bound for broadcast in radio networks, SIAM J. Comput. 27 (1998) 702–712.